



# AZ-104T00A

## Module 07:

### Azure Storage



# Module Overview

- Lesson 01: Storage Accounts
- Lesson 02: Blob Storage
- Lesson 03: Storage Security
- Lesson 04: Azure Files and File Sync
- Lesson 05: Managing Storage
- Lesson 06: Module 07 Lab and Review

# Lesson 01: Storage Accounts



# Storage Accounts Overview

- Azure Storage
- Azure Storage Services
- Storage Account Kinds
- Replication Strategies
- Accessing Storage
- Securing Storage Endpoints
- Demonstration – Securing a Storage Endpoint

# Azure Storage

- A service that you can use to store files, messages, tables, and other types of information
- Durable, secure, scalable, managed, accessible
- Manage data with multiple storage accounts
- Three categories of Azure storage:
  - Storage for virtual machines – Disks and File Shares
  - Unstructured data – Blobs and Data Lake Store
  - Structured data - Tables, Cosmos DB, and Azure SQL DB
- Standard storage backed by magnetic drives (HDD) is lowest cost
- Premium storage backed by solid state drives (SSD)

# Azure Storage Services

- **Azure Containers:** A massively scalable object store for text and binary data
- **Azure Files:** Managed file shares for cloud or on-premises deployments
- **Azure Tables:** A NoSQL store for schemaless storage of structured data
- **Azure Queues:** A messaging store for reliable messaging between application components



## Containers

Scalable, cost-effective storage for unstructured data

[Learn more](#)



## File shares

Serverless SMB file shares

[Learn more](#)



## Tables

Tabular data storage

[Learn more](#)



## Queues

Effectively scale apps according to traffic

[Learn more](#)

# Storage Account Kinds

| Storage account type           | Supported services                       | Supported tiers   | Replication options                                      |
|--------------------------------|------------------------------------------|-------------------|----------------------------------------------------------|
| BlobStorage                    | Blob (block blobs and append blobs only) | Standard          | LRS, GRS, RA-GRS                                         |
| Storage (general purpose v1)   | Blob, File, Queue, Table, and Disk       | Standard, Premium | LRS, GRS, RA-GRS                                         |
| StorageV2 (general purpose v2) | Blob, File, Queue, Table, and Disk       | Standard, Premium | LRS, GRS, RA-GRS, ZRS, ZGRS (preview), RA-ZGRS (preview) |
| Block blob storage             | Blob (block blobs and append blobs only) | Premium           | LRS, ZRS (limited regions)                               |
| File Storage                   | Files only                               | Premium           | LRS, ZRS (limited regions)                               |

✓ All storage accounts are encrypted using Storage Service Encryption (SSE) for data at rest

# Replication Strategies

| Data Replication Options                         | Description                                                                                                         |
|--------------------------------------------------|---------------------------------------------------------------------------------------------------------------------|
| Locally redundant storage (LRS)                  | Data is replicated three time within a single facility in a single region.                                          |
| Zone-redundant storage (ZRS)                     | Data is replicated three times across two to three facilities, either within a single region or across two regions. |
| Geo-redundant storage (GRS)                      | Data is replicated three times within the primary region and replicated three times to the regions pair.            |
| Read access geo-redundant storage (RA-GRS)       | Data is replicated three times within the primary region and replicated with read-access to the region pair.        |
| Geo-zone-redundant storage (GZRS)                | Data is replicated across three Availability Zones and replicated to the region pair.                               |
| Read-access Geo-zone-redundant storage (RA-GZRS) | Data is replicated across three Availability Zones and replicated with read-access to the region pair.              |



# Accessing Storage

| CNAME record      | Target                             |
|-------------------|------------------------------------|
| blobs.contoso.com | contosoblobs.blob.core.windows.net |

- Every object has a unique URL address
- The storage account name forms the subdomain of that address
- The subdomain and domain name forms an *endpoint*
  - **Container service:** `http://mystorageaccount.blob.core.windows.net`
  - **Table service:** `http://mystorageaccount.table.core.windows.net`
  - **Queue service:** `http://mystorageaccount.queue.core.windows.net`
  - **File service:** `http://mystorageaccount.file.core.windows.net`
- If you prefer you can configure a custom domain name

# Securing Storage Account Endpoints

storage987123 | Firewalls and virtual networks

Storage account

Search (Ctrl+ /)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Data transfer

Events

Storage Explorer (preview)

Save Discard Refresh

Allow access from

All networks

Selected networks

Configure network security for your storage accounts. [Learn more.](#)

Virtual networks

Secure your storage account with virtual networks. [+ Add existing virtual network](#) [+ Add new virtual network](#)

| Virtual Network | Subnet   | Address range | Endpoint Status | Resource Group |
|-----------------|----------|---------------|-----------------|----------------|
| ▼ vnet01        | 1        |               |                 | Demo           |
|                 | subnet01 | 10.1.0.0/24   | ✓ Enabled       | Demo           |

- Firewalls and Virtual Networks allows for restricting access to the Storage Account from specific Subnets on Virtual Networks
- Subnets and Virtual Networks must exist in the same Azure Region or Region Pair as the Storage Account

# Demonstration – Securing a Storage Endpoint

- Create a storage account
- Upload a file to the storage account
- Create a subnet service endpoint
- Secure the storage to the service endpoint
- Test the storage endpoint

# Lesson 02: Blob Storage

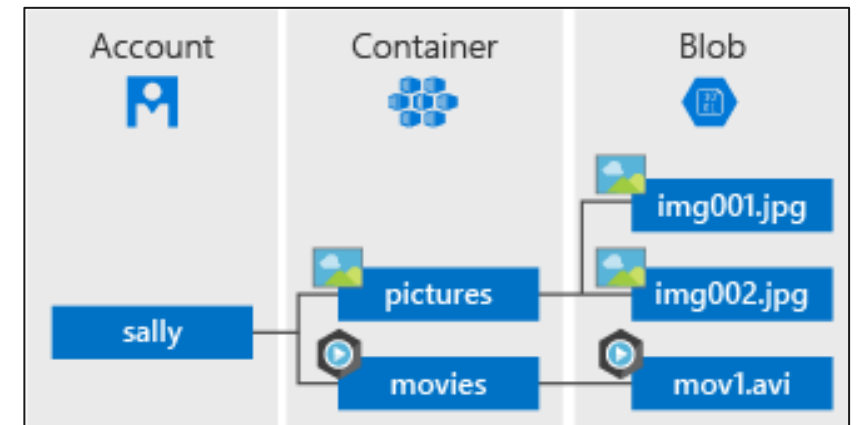


# Blob Storage Overview

- Blob Storage
- Blob Containers
- Blob Access Tiers
- Blob Lifecycle Management
- Uploading Blobs
- Storage Pricing
- Demonstration – Blob Storage

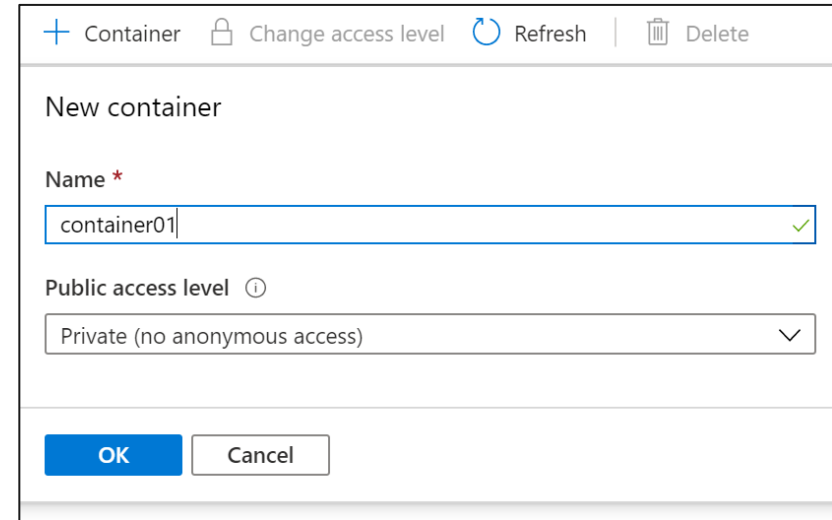
# Blob Storage

- Stores unstructured data in the cloud
- Can store any type of text or binary data
- Also referred to as *object storage*
- Common uses:
  - Serving images or documents directly to a browser
  - Storing files for distributed access
  - Streaming video and audio
  - Storing data for backup and restore, disaster recovery, archiving
  - Storing data for analysis by an on-premises or Azure-hosted service

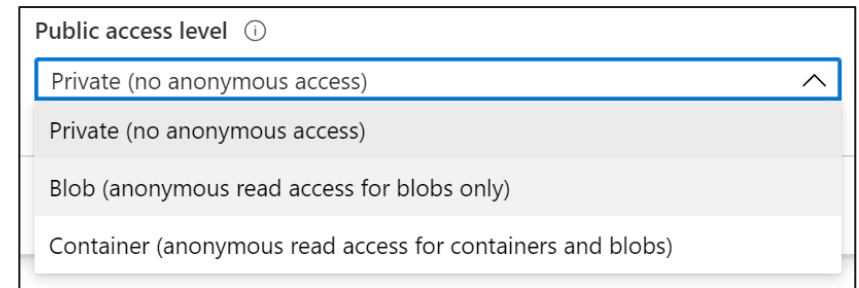


# Blob Containers

- All blobs must be in a container
- Accounts have unlimited containers
- Containers can have unlimited blobs
- Private blobs - no anonymous access
- Blob access - anonymous public read access for blobs only
- Container access - anonymous public read and list access to the entire container, including the blobs



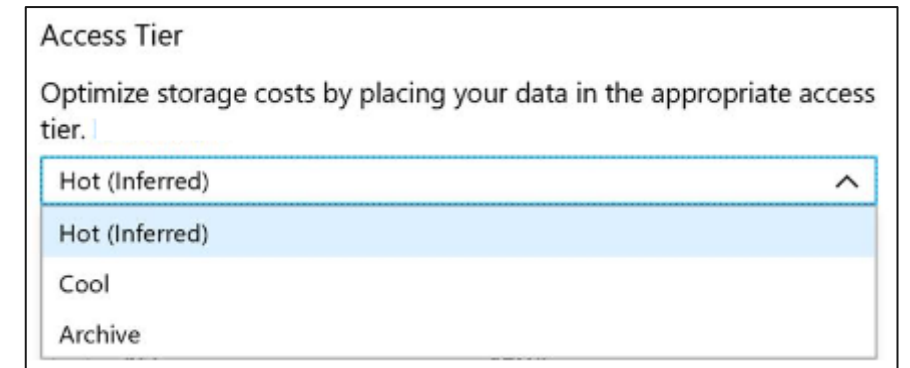
The screenshot shows the 'New container' dialog box. At the top, there are icons for '+ Container', 'Change access level', 'Refresh', and 'Delete'. The main section is titled 'New container'. It has a 'Name' field with a red asterisk, containing the text 'container01' and a green checkmark. Below it is a 'Public access level' dropdown menu with a circled 'i' icon, currently set to 'Private (no anonymous access)'. At the bottom, there are 'OK' and 'Cancel' buttons.



The screenshot shows the 'Public access level' dropdown menu expanded. The title 'Public access level' is at the top with a circled 'i' icon. The dropdown list contains four options: 'Private (no anonymous access)' (highlighted with a blue border), 'Private (no anonymous access)', 'Blob (anonymous read access for blobs only)', and 'Container (anonymous read access for containers and blobs)'.

# Blob Access Tiers

- **Hot tier** - Optimized for frequent access of objects in the storage account
- **Cool tier** - Optimized for storing large amounts of data that is infrequently accessed and stored for at least 30 days
- **Archive** - Optimized for data that can tolerate several hours of retrieval latency and will remain in the Archive tier for at least 180 days



The screenshot shows a dropdown menu titled 'Access Tier' with the instruction 'Optimize storage costs by placing your data in the appropriate access tier.' The menu is open, showing four options: 'Hot (Inferred)' (selected), 'Hot (Inferred)', 'Cool', and 'Archive'. The first 'Hot (Inferred)' option is highlighted with a blue background.

| Access Tier    |
|----------------|
| Hot (Inferred) |
| Hot (Inferred) |
| Cool           |
| Archive        |

✓ You can switch between these access tiers at any time.



# Blob Lifecycle Management

Blob Lifecycle Management allows for:

- Transitioning of blobs to a cooler storage tier to optimize for performance and cost
- Delete blobs at the end of their lifecycle
- Apply rules to filtered paths in the Storage Account

**Rule name \***

**Blobs**

☒ Move blob to cool storage  
Days after last modification

☒ Move blob to archive storage  
Days after last modification

☒ Delete blob  
Days after last modification

**Snapshots**

☒ Delete snapshot  
Days after blob is created

# Uploading Blobs

- Block blobs (default) - useful for storing text or binary files
  - Page blobs - More efficient for frequent read/write operations
  - Append blobs - useful for logging scenarios
  - Access tier – select either Hot, Cool, or Archive
- ✓ You cannot change a blob type once it has been created

Upload blob

container01/

Files ⓘ

"Cloud.jpg" 

☐ Overwrite if files already exist

^ Advanced

Authentication type ⓘ

Azure AD user account

Account key

Blob type ⓘ

Block blob 

☒ Upload .vhd files as page blobs (recommended)

Block size ⓘ

4 MB 

Access tier ⓘ

Hot (Inferred) 

Upload to folder

Upload

# Storage Pricing

- Storage costs
- Blob storage
- Data access costs
- Transaction costs
- Geo-Replication data transfer costs
- Outbound data transfer costs
- Changing the storage tier

| Block Blobs                                                                                                                                    | Files                                                                                                                                                                                  |
|------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Scalable object storage for documents, videos, pictures, and unstructured text or binary data. Choose from Hot, Cool, or Archive tiers.</p> | <p>Fully managed file shares in the cloud, accessible via standard Server Message Block (SMB) protocol. Enables sharing files between applications using Windows APIs or REST API.</p> |
| <p>Prices for locally redundant storage (LRS) Archive Block Blob start from:</p>                                                               | <p>Prices for LRS File storage start from:</p>                                                                                                                                         |
| <p><b>\$0.002</b>/GB per month</p>                                                                                                             | <p><b>\$0.06</b>/GB per month</p>                                                                                                                                                      |
| <p><a href="#">See Pricing &gt;</a></p>                                                                                                        | <p><a href="#">See Pricing &gt;</a></p>                                                                                                                                                |

# Demonstration – Blob Storage

- Create a container
- Upload a block blob
- Download a block blob

# Lesson 03: Storage Security



# Storage Security Overview

- Storage Security
- Shared Access Signatures
- URI and SAS Parameters
- Demonstration – SAS (Portal)
- Storage Service Encryption
- Customer Managed Keys
- Storage Security Best Practices

# Storage Security

- Storage Encryption Services
- Authentication with Azure AD and RBAC
- Client-side encryption, HTTPS, and SMB 3.0 for data in transit
- Azure disk encryption
- Shared Access Signatures – delegated access
- Shared Key – encrypted signature string
- Anonymous access to containers and blobs

# Shared Access Signatures

- Provides delegated access to resources
- Grants access to clients without sharing your storage account keys
- The account SAS delegates access to resources in one or more of the storage services
- The service SAS delegates access to a resource in just one of the storage services

★ Permissions ⓘ

Read ▼

Start and expiry date/time ⓘ

Start

2019-02-27  7:32:03 AM

2019-02-27  3:32:03 PM

(UTC-08:00) --- Current Time Zone --- ▼

(UTC-08:00) --- Current Time Zone --- ▼

Allowed IP addresses ⓘ

for example, 168.1.5.65 or 168.1.5.65-168.1.5.70

Allowed protocols ⓘ

☒ HTTPS ☐ HTTP

Signing key ⓘ

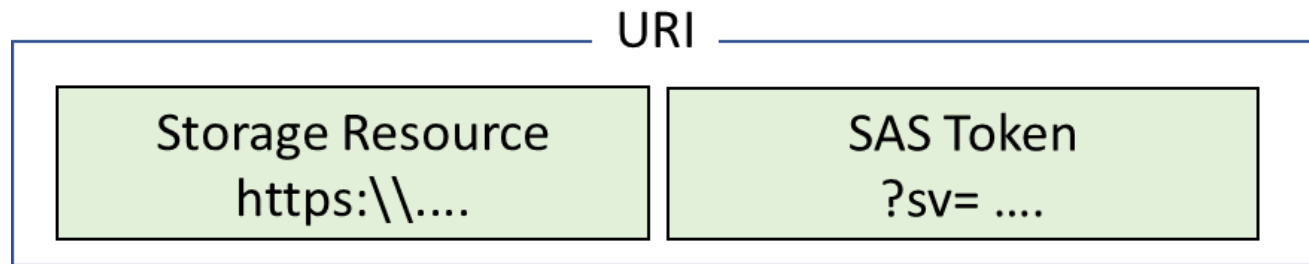
Key 1 ▼

Generate blob SAS token and URL



# URI and SAS Parameters

- A SAS is a signed URI that points to one or more storage resources
- Consists of a storage resource URI and the SAS token



<https://myaccount.blob.core.windows.net/?restype=service&comp=properties&sv=2015-04-05&ss=bf&srt=s&st=2015-04-29T22%3A18%3A26Z&se=2015-04-30T02%3A23%3A26Z&sr=b&sp=rw&sip=168.1.5.60-168.1.5.70&spr=https&sig=F%6GRVAZ5Cdj2Pw4txxxxx>

- Includes parameters for resource URI, storage services version, services, resource types, start time, expiry time, resource, permissions, IP range, protocol, signature

# Demonstration – SAS (Portal)

- Create a SAS at the service level
- Create a SAS at the account level

# Storage Service Encryption

- Protects your data for security and compliance
  - Automatically encrypts and decrypts your data
  - Encrypted through 256-bit AES encryption
  - Is enabled for all new and existing storage accounts and cannot be disabled
  - Is transparent to users
- ✓ You can use your own key (next topic)

**Encryption**

 Save  Discard

Storage service encryption protects your data at rest. Azure Storage encrypts your data as it's written in our datacenters, and automatically decrypts it for you as you access it.

By default, data in the storage account is encrypted using Microsoft Managed Keys. You may choose to bring your own key.

Please note that after enabling Storage Service Encryption, only new data will be encrypted, and any existing files in this storage account will retroactively get encrypted by a background encryption process.

[Learn More about Azure Storage Encryption](#) 

**Encryption type**

☒ Microsoft Managed Keys

☐ Customer Managed Keys

# Customer Managed Keys

- Use the Azure Key Vault to manage your encryption keys
- Create your own encryption keys and store them in a key vault
- Use Azure Key Vault's APIs to generate encryption keys
- Custom keys give you more flexibility and control

## Encryption type

☐ Microsoft Managed Keys

☒ Customer Managed Keys

**i** The storage account named 'storage987123' will be granted access to the selected key vault. Both soft delete and purge protection will be enabled on the key vault and cannot be disabled. [Learn more about customer managed keys](#) 

## Encryption key

☐ Enter key URI

☒ Select from Key vault

## Key vault and key \*

Key vault: keyvault987123

Key: storagekey

[Select a key vault and key](#)

# Storage Best Practices

- Always use HTTPS to create or distribute an SAS
- Reference stored access policies where possible
- Use near-term expiration times on an ad hoc SAS
- Have clients automatically renew the SAS if necessary
- Be careful with SAS start time
- Be specific with the resource to be accessed
- Understand that your account will be billed for any usage
- Validate data written using SAS
- Don't assume SAS is always the correct choice
- Use Storage Analytics to monitor your application

# Lesson 04: Azure Files and File Sync



# Azure Files and File Sync Overview

- Files vs Blobs
- Managing File Shares
- File Share Snapshots
- Demonstration – File Shares
- Azure File Sync
- Azure File Sync Components
- File Sync Steps

# Files vs Blobs

| Feature     | Description                                                                                                                                       | When to use                                                                                                                                                                                                                                                      |
|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Azure Files | SMB interface, client libraries, and a REST interface that allows access from anywhere to stored files.                                           | <ul style="list-style-type: none"><li>• Lift and shift an application to the cloud.</li><li>• Store shared data across multiple virtual machines.</li><li>• Store development and debugging tools that need to be accessed from many virtual machines.</li></ul> |
| Azure Blobs | Client libraries and a REST interface that allows unstructured data (flat namespace) to be stored and accessed at a massive scale in block blobs. | <ul style="list-style-type: none"><li>• Support streaming and random-access scenarios.</li><li>• Access application data from anywhere.</li></ul>                                                                                                                |



# Managing File Shares

- File share quotas
- Windows – ensure port 445 is open
- Linux – mount the drive
- MacOS – mount the drive
- Secure transfer required - SMB 3.0 encryption

Windows Linux macOS

Drive letter

Z

To connect to this Azure file share from Windows, run these PowerShell commands from a normal (not elevated) PowerShell terminal:

```
$connectTestResult = Test-NetConnection -  
ComputerName storage987123.file.core.windows.net -  
Port 445  
if ($connectTestResult.TcpTestSucceeded) {  
    # Save the password so the drive will persist on reboot  
    cmd.exe /C "cmdkey  
/add:"storage987123.file.core.windows.net"
```

This script will check to see if this storage account is accessible via TCP port 445, which is the port SMB uses. If port 445 is available, your Azure file share will be persistently mounted. Your organization or internet service provider (ISP) may block port 445, however you may use Azure [Point-to-Site \(P2S\) VPN](#), Azure [Site-to-Site \(S2S\) VPN](#), or [ExpressRoute](#) to tunnel SMB traffic to your Azure file share over a different port.

# File Share Snapshots

|  Add snapshot  Refresh  Delete |                       |           |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|-----------|
| Name                                                                                                                                                                                                                                                                                | Date created          | Initiator |
| <input type="checkbox"/> 2020-03-12T00:58:38.0000000Z                                                                                                                                                                                                                               | 3/11/2020, 8:58:38 PM | -         |

- Incremental snapshot that captures the share state at a point in time
- Is read-only copy of your data
- Snapshot at the file share level, and restore at the file level
- Uses:
  - Protection against application error and data corruption.
  - Protection against accidental deletions or unintended changes.
  - General backup purposes.

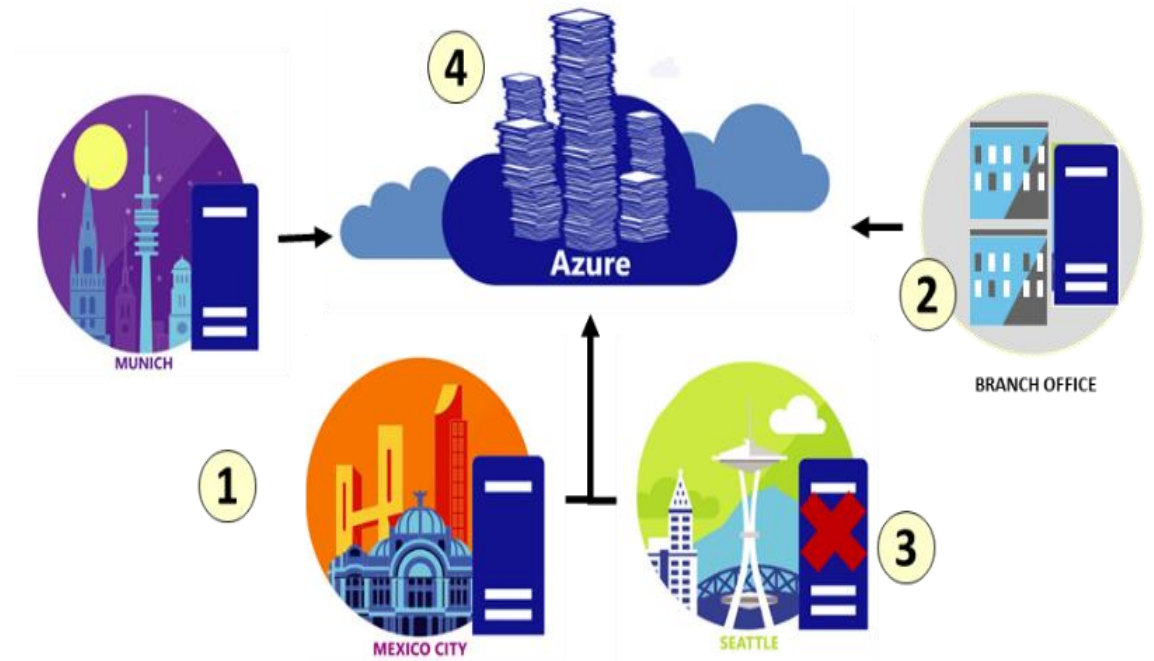
# Demonstration – File Shares

- Create a file share and upload a file
- Manage snapshots
- Create a file share (PowerShell)
- Mount a file share (PowerShell)

# Azure File Sync

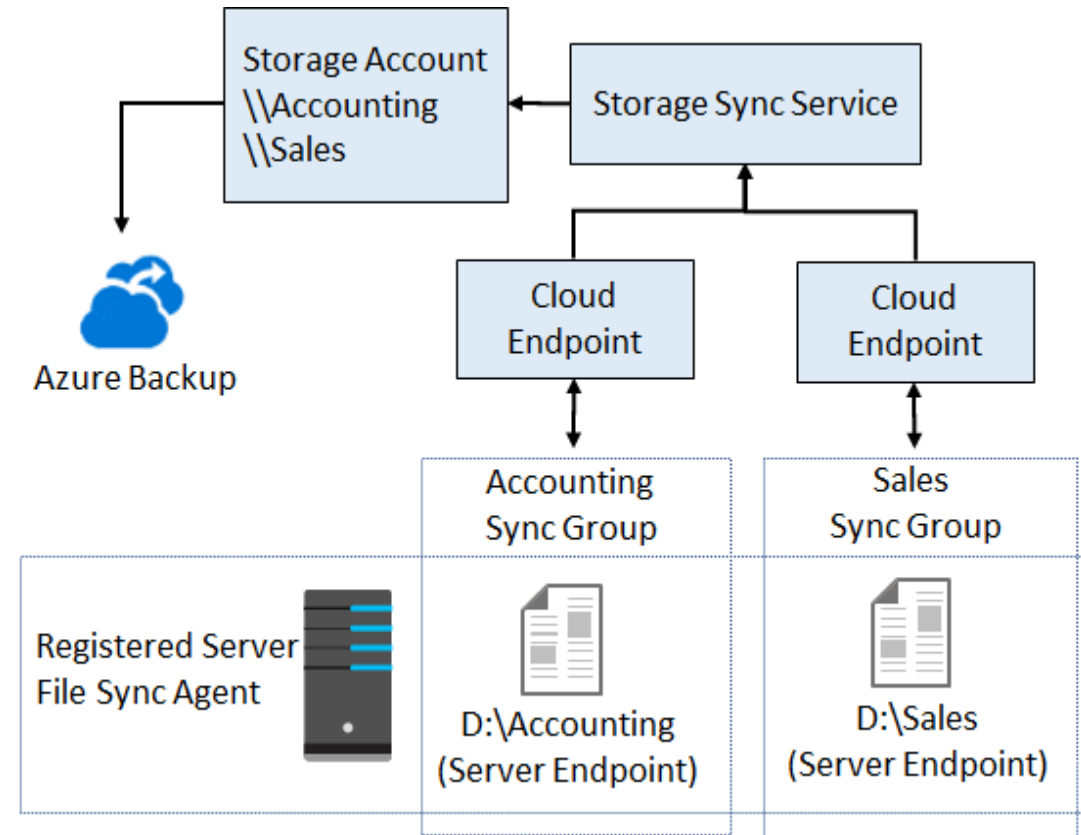
Centralize file shares in Azure Files

1. Lift and shift
2. Branch Office backups
3. Backup and Disaster Recovery
4. File Archiving

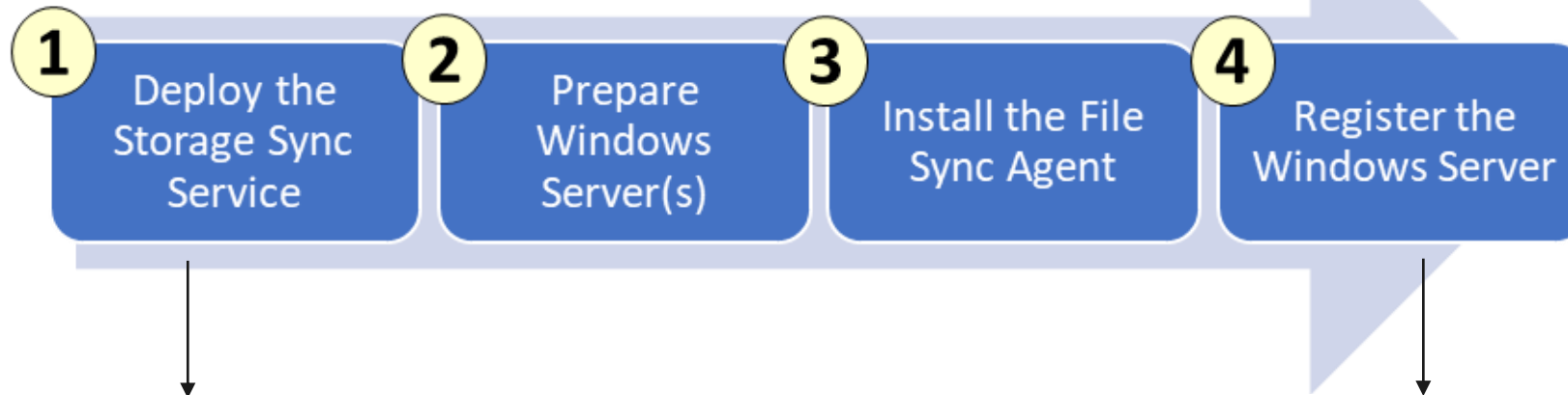


# File Sync Components

- The Storage Sync Service is the top-level resource.
- The **registered server** object represents a trust relationship between your server (or cluster) and the Storage Sync Service
- The **Azure File Sync agent** is a downloadable package that enables Windows Server to be synced with an Azure file share
- A **server endpoint** represents a specific location on a registered server, such as a folder
- A **cloud endpoint** is an Azure file share
- A **sync group** defines which files are kept in sync



# File Sync Steps



Home > Deploy Storage Sync

### Deploy Storage Sync

\* Name  
StorageSync1 ✓

\* Subscription  
Visual Studio Enterprise ▼

\* Resource group  
ASH ▼

[Create new](#)

\* Location  
South Central US ▼

[Create](#) [Automation options](#)

### Microsoft Azure File Sync - Server Registration

#### Choose a Storage Sync Service

Azure Subscription  
▼

Resource Group  
▼

Storage Sync Service  
▼

[Register](#)

# Lesson 05: Managing Storage



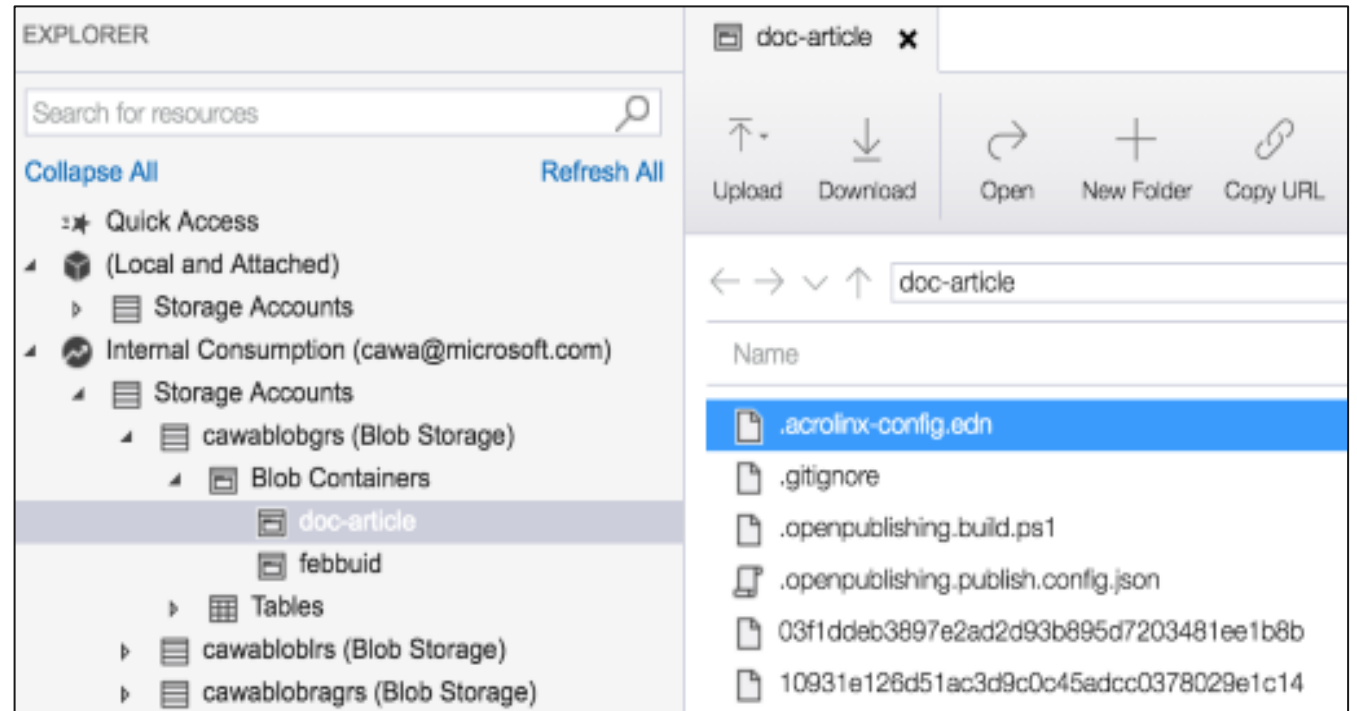
# Managing Storage Overview

- Storage Explorer
- Import and Export Service
- Data Box
- AzCopy
- Data Transfer Tool Selection
- Demonstration – Storage Explorer
- Demonstration - AzCopy



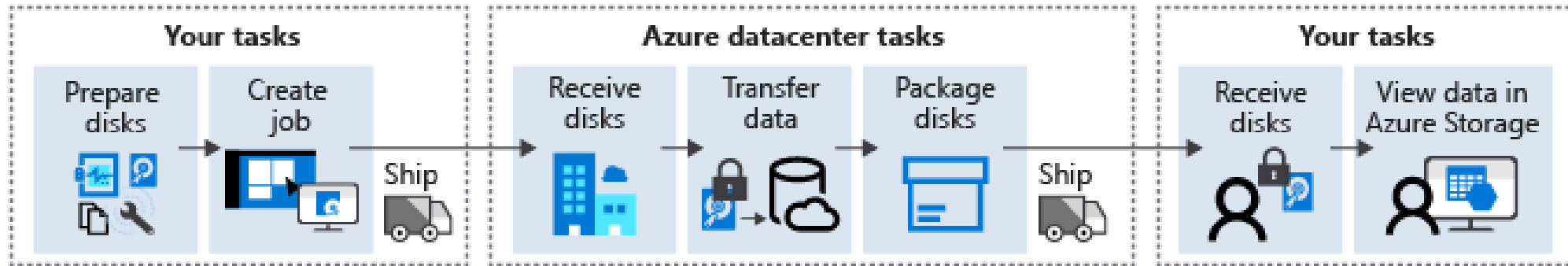
# Storage Explorer

- Access multiple accounts and subscriptions
- Create, delete, view, edit storage resources
- View and edit Blob, Queue, Table, File, Cosmos DB storage and Data Lake Storage
- Obtain shared access signature (SAS) keys
- Available for Windows, Mac, and Linux



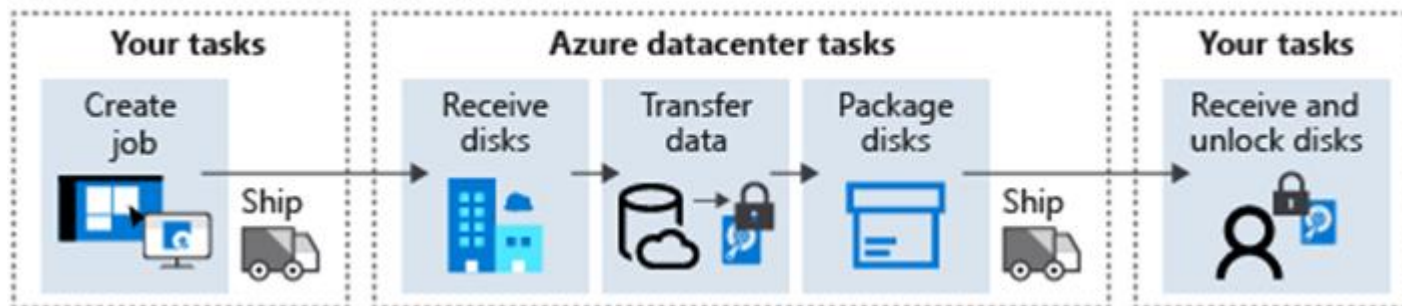
# Import and Export Service

- Import jobs move large amounts of data to Azure blob storage or files

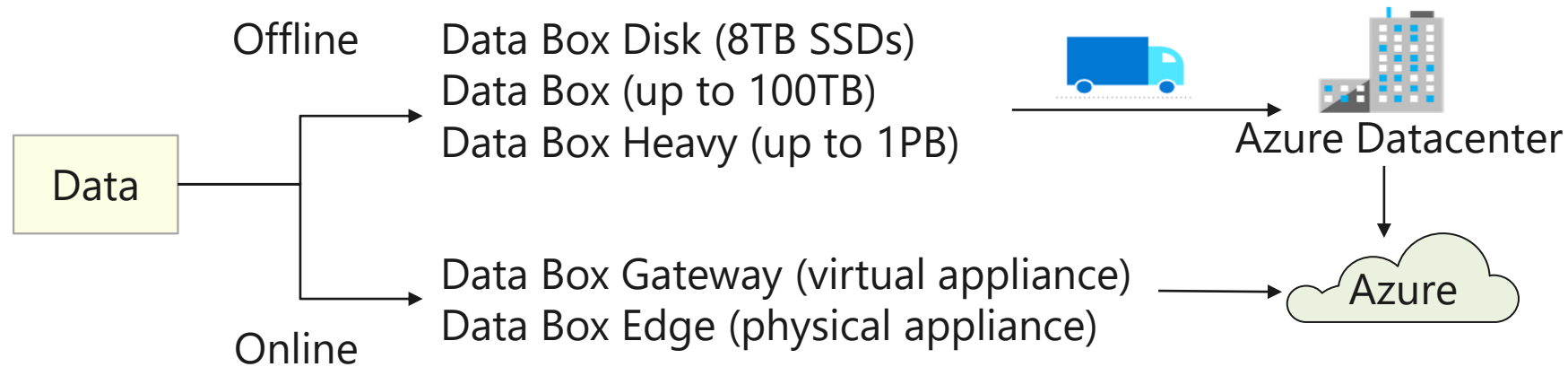


- Export jobs move large amounts of data from Azure storage (not files)

Export data with Azure Import/Export



# Data Box



- Easy, secure, fast large volume data transfer
- **Offline** usage - one-time migration, incremental transfer, periodic updates
- **Online** usage – cloud archival, data aggregation, integration with on-premises workloads, pre-process data (Edge), inference Azure Machine Learning (Edge)

# AzCopy

`AzCopy /Source:<source> /Dest:<destination> [Options]`

- Command-line utility
- Available on Windows, Linux, and MacOS
- Designed for copying data to and from Azure Blob, File, and Table storage
- Authentication options include Active Directory or SAS token
- Example 1: Copy a Blob storage account to another account
- Example 2: List/Remove files and blobs (wildcard support)

# Data Transfer Tool Selection

| Dataset                               | Network bandwidth                                                                                       | Solution to use                                                                                                         |
|---------------------------------------|---------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|
| Large dataset                         | Low-bandwidth network or direct connectivity to on-premises storage is limited by organization policies | Azure Import/Export for export; Data Box Disk or Data Box for import where supported; otherwise use Azure Import/Export |
| Large dataset                         | High-bandwidth network: 1 gigabit per second (Gbps) - 100 Gbps                                          | AzCopy for online transfers; or to import data, Azure Data Box Edge, or Azure Data Box Gateway                          |
| Large dataset                         | Moderate-bandwidth network: 100 megabits per second (Mbps) - 1 Gbps                                     | Azure Import/Export for export or Azure Data Box family for import where supported                                      |
| Small dataset: a few GBs to a few TBs | Low to moderate-bandwidth network: up to 1 Gbps                                                         | If transferring only a few files, use Azure Storage Explorer, Azure portal, AzCopy, or AZ CLI                           |

# Demonstration – Storage Explorer

- Download and install Storage Explorer
- Connect to an Azure subscription
- Attach an Azure storage account
- Generate a SAS connection string for the account you want to share
- Attach to a storage account by using a SAS Connection string

# Demonstration - AzCopy

- Install the AzCopy tool
- Explore the help
- Download a blob from Blob storage to the file system
- Upload files to Azure blob storage

# Lesson 06: Module 07 Labs and Review





# Lab 07 – Manage Azure Storage

## Lab scenario

You need to evaluate the use of Azure storage for storing files residing currently in on-premises data stores. While many of these files are not accessed frequently, there are some exceptions. You would like to minimize cost of storage by placing less frequently accessed files in lower-priced storage tiers. You also plan to explore different protection mechanisms that Azure Storage offers, including network access, authentication, authorization, and replication. Finally, you want to determine to what extent Azure Files service might be suitable for hosting your on-premises file shares.

## Objectives

- Task 1: Provision the lab environment
- Task 2: Create and configure Azure Storage accounts
- Task 3: Manage blob storage
- Task 4: Manage authentication and authorization for Azure Storage
- Task 5: Create and configure an Azure Files shares
- Task 6: Manage network access for Azure Storage

# Module Review

- Module Review Questions
- Microsoft Learn Modules ([docs.microsoft.com/Learn](https://docs.microsoft.com/Learn))
  - Create an Azure Storage account
  - Secure your Azure Storage
  - Optimize storage performance and costs using Blob storage tiers
  - Make your application storage highly available with read-access geo-redundant storage
  - Copy and move blobs from one container or storage account to another from the command line and in code
  - Move large amounts of data to the cloud by using Azure Data Box family
  - Monitor, diagnose, and troubleshoot your Azure storage